

BEST AVAILABLE COPY
PATENT ABSTRACTS OF JAPAN

(11)Publication number : 2000-175158
 (43)Date of publication of application : 23.06.2000

(51)Int.CI. H04N 5/93
 G11B 20/10
 H04N 5/765
 H04N 5/781
 H04N 5/91

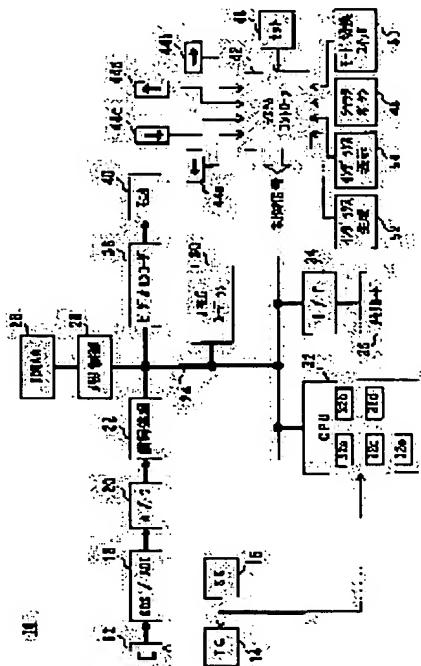
(21)Application number : 10-349996 (71)Applicant : SANYO ELECTRIC CO LTD
 (22)Date of filing : 09.12.1998 (72)Inventor : KAKU JUNYA

(54) IMAGE REPRODUCTION DEVICE

(57)Abstract:

PROBLEM TO BE SOLVED: To provide an image reproduction device that can access a desired image in a short time.

SOLUTION: A plurality of still picture files consecutive to each other are contained in a moving picture file stored in an SDRAM 28 and a moving picture is displayed on a monitor 40 by reproducing them continuously. When an index display button 54 is depressed during reproduction of the moving picture, a CPU 32 reads an index file from a memory card 36 and reproduces a plurality of thumbnail pictures contained in the file. The monitor 40 displays a plurality of thumbnail pictures in place of the moving picture. When an operator selects a desired thumbnail picture among the displayed thumbnail pictures, a still picture file corresponding to the selected thumbnail picture is reproduced. As a result, the same still picture as the selected thumbnail picture is displayed on a full screen of the monitor 40. When a set key 46 is depressed, a moving picture in succession to this still picture is reproduced. The operator can access a desired recording part in a short time by using the index file.



LEGAL STATUS

[Date of request for examination] 22.10.1999

[Date of sending the examiner's decision of rejection]

[Kind of final disposal of application other than the examiner's decision of rejection or application converted registration]

[Date of final disposal for application]

[Patent number] 3269037

[Date of registration] 18.01.2002

[Number of appeal against examiner's decision of rejection]

[Date of requesting appeal against examiner's decision of rejection]

[Date of extinction of right]

Copyright (C); 1998,2003 Japan Patent Office

【特許請求の範囲】

【請求項1】複数の静止画像を含む動画像ファイルを再生して動画像をモニタに出力する画像再生装置において、前記複数の静止画像から所望の静止画像を特定する第1特定手段、および前記所望の静止画像のインデックス情報を生成する作成手段を備えることを特徴とする、画像再生装置。

【請求項2】前記第1特定手段は、前記動画像の出力中に前記インデックス情報の作成指示を受け付ける作成指示受付手段、および前記作成指示の受付タイミングに基づいて前記所望の静止画像を特定する第1静止画像特定手段を含む、請求項1記載の画像再生装置。

【請求項3】前記第1特定手段は、前記所望の静止画像を前記モニタに継続して出力する第1静止画像出力手段をさらに含む、請求項2記載の画像再生装置。

【請求項4】前記作成手段は、前記所望の静止画像に基づいてサムネイル画像を生成するサムネイル画像生成手段、前記所望の静止画像の位置情報を検出する位置情報検出手段、および前記サムネイル画像と前記位置情報を関連付けて前記インデックス情報を生成するインデックス情報作成手段を含む、請求項1ないし3のいずれかに記載の画像再生装置。

【請求項5】前記位置情報は、前記所望の静止画像の前記動画像ファイルにおけるフレーム番号である、請求項4記載の画像再生装置。

【請求項6】前記所望の静止画像は複数存在し、前記インデックス情報に基づいて1つの静止画像を特定する第2特定手段、および前記第2特定手段によって特定された特定静止画像を前記動画像ファイルから再生する第1再生手段をさらに備える、請求項1ないし3のいずれかに記載の画像再生装置。

【請求項7】前記インデックス情報は前記所望の静止画像に対応する複数のサムネイル画像と前記所望の静止画像の位置情報を含み、前記第2特定手段は、前記インデックス情報の再生指示を受け付ける第1再生指示受付手段、前記インデックス情報の再生指示に応答して前記複数のサムネイル画像をモニタに出力するサムネイル画像出力手段、1つのサムネイル画像の選択を受け付ける選択受付手段、および前記1つのサムネイル画像に対応する位置情報を前記インデックス情報に基づいて検出する位置情報検出手段を含む、請求項6記載の画像再生装置。

【請求項8】前記第1再生手段は、前記位置情報検出手段によって検出された位置情報に対応する静止画像を前記モニタに継続して出力する第2静止画像出力手段を含む、請求項7記載の画像再生装置。

【請求項9】前記位置情報は、前記静止画像の前記動画像ファイルにおけるフレーム番号である、請求項7または8記載の画像再生装置。

【請求項10】前記特定静止画像に続く動画像を再生する第2再生手段をさらに備える、請求項6ないし9のいずれかに記載の画像再生装置。

【請求項11】前記第2再生手段は、前記特定静止画像の再生後に動画像の再生指示を受け付ける第2再生指示受付手段、および前記動画像の再生指示に応答して前記特定静止画像に続く動画像を出力する動画像出力手段を含む、請求項10記載の画像再生装置。

【請求項12】複数の静止画像を含む動画像ファイルを再生して動画像をモニタに出力する画像再生装置において、インデックス情報に基づいて所望の静止画像を特定する特定手段、および前記所望の静止画像を前記動画像ファイルから再生する再生手段を備える、画像再生装置。

【請求項13】請求項1ないし12のいずれかに記載の画像再生装置を備える、デジタルカメラ。

【発明の詳細な説明】

【0001】

【産業上の利用分野】この発明は、画像再生装置に関する、特にたとえばデジタルカメラに適用され、複数の静止画像を含む動画像ファイルを再生する、画像再生装置に関する。

【0002】

【従来の技術】従来のこの種のデジタルカメラでは、動画像ファイルは常に先頭から再生されていた。つまり、モニタに表示したい画像が動画像ファイルの中盤に記録されているときでも、動画像の再生は先頭から開始されていた。

【0003】

【発明が解決しようとする課題】このため、所望の画像が再生されるまで、余分な時間がかかるという問題があった。それゆえに、この発明の主たる目的は、短時間で所望の画像にアクセスできる、画像再生装置を提供することである。

【0004】

【課題を解決するための手段】第1の発明は、複数の静止画像を含む動画像ファイルを再生して動画像をモニタに出力する画像再生装置において、複数の静止画像から所望の静止画像を特定する第1特定手段、および所望の静止画像のインデックス情報を生成する作成手段を備えることを特徴とする、画像再生装置である。

【0005】

【0005】第2の発明は、複数の静止画像を含む動画像ファイルを再生して動画像をモニタに出力する画像再生装置において、インデックス情報に基づいて所望の静止画像を特定する特定手段、および所望の静止画像を動画像ファイルから再生する再生手段を備える、画像再生装置である。

【0006】

【作用】第1の発明では、第1特定手段が動画像ファイルに含まれる複数の静止画像から所望の静止画像を特定

すると、作成手段が所望の静止画像のインデックス情報を作成する。この発明のある局面では、第1特定手段は次のようにして所望の静止画像を特定する。つまり、動画像がモニタに出力されている間に、作成指示受付手段がインデックス情報を作成指示を受け付ける。そして、作成指示の受付タイミングに基づいて、第1静止画像特定手段が所望の静止画像を特定する。

【0007】この発明のある実施例では、所望の静止画像が特定されると、同じ静止画像がモニタに継続して出力される。つまり、モニタの表示は動画像から所望の静止画像に切り換わる。この発明の他の局面では、作成手段は次のようにしてインデックス情報を作成する。つまり、サムネイル画像生成手段が所望の静止画像に基づいてサムネイル画像を生成し、位置情報検出手段が所望の静止画像の位置情報を検出する。そして、インデックス情報作成手段が、サムネイル画像と位置情報とを関連付けてインデックス情報を作成する。この発明のある実施例では、位置情報は、所望の静止画像の動画像ファイルにおけるフレーム番号である。

【0008】この発明の他の局面では、所望の静止画像は複数存在する。第2特定手段は、インデックス情報に基づいて1つの静止画像を特定し、第1再生手段は、第2特定手段によって特定された静止画像を動画像ファイルから再生する。この発明のある実施例では、インデックス情報は所望の静止画像にそれぞれ対応する複数のサムネイル画像と所望の静止画像のそれぞれの位置情報を含む。第2特定手段は、以下のようにして静止画像を特定する。つまり、再生指示受付手段が上述のようなインデックス情報の再生指示受け付け、サムネイル画像出力手段が再生指示に応答して複数のサムネイル画像をモニタに出力する。選択受付手段は1つのサムネイル画像の選択を受け付け、位置情報検出手段は、選択されたサムネイル画像に対応する位置情報をインデックス情報に基づいて検出する。この結果、位置情報に対応する静止画像が特定される。

【0009】第1再生手段に含まれる第2静止画像出力手段は、次のようにして特定された静止画像をモニタに継続して出力する。なお、位置情報は、静止画像の動画像ファイルにおけるフレーム番号である。この発明の他の実施例では、第2再生手段が第1再生手段によって再生された静止画像に続く動画像を再生する。つまり、静止画像の再生後に動画像の再生指示を受け付け、このような再生指示が与えられると、上述の動画像をモニタに出力する。

【0010】第2の発明では、特定手段がインデックス情報に基づいて所望の静止画像を特定し、再生手段が特定された所望の静止画像を動画像ファイルから再生する。

【0011】

【発明の効果】第1の発明によれば、動画像ファイルに

含まれる複数の静止画像から所望の静止画像を特定し、特定した静止画像のインデックス情報を作成するようにしたため、このインデックス情報を利用することで所望の静止画像に短時間でアクセスすることができる。

【0012】第2の発明によれば、インデックス情報に基づいて特定された所望の静止画像を動画像ファイルから再生するようにしたため、所望の静止画像に短時間でアクセスすることができる。この発明の上述の目的、その他の目的、特徴および利点は、図面を参照して行う以下の実施例の詳細な説明から一層明らかとなろう。

【0013】

【実施例】図1を参照して、この実施例のデジタルカメラ10はCCDイメージヤ12を含む。CCDイメージヤ12の前面には色フィルタ（図示せず）が装着され、被写体の光像はこの色フィルタを介してCCDイメージヤ12に照射される。モード設定スイッチ50を“カメラ”側に切り換えると、システムコントローラ42がカメラモードを設定する。タイミングジェネレータ（TG）14は、シグナルジェネレータ（SG）16から出力される垂直同期信号および水平同期信号に基づいてタイミング信号を生成し、CCDイメージヤ12をプログレッシブスキャン方式で駆動する。この結果、被写体のカメラ信号がCCDイメージヤ12から出力される。出力されたカメラ信号は、CDS/AGC回路18で周知のノイズ除去およびレベル調整を施され、その後、A/D変換器16によってデジタル信号であるカメラデータに変換される。信号処理回路22は、A/D変換器16から出力されたカメラデータにYUV変換を施し、YUVデータを生成する。

【0014】生成されたYUVデータはバス24を介してメモリ制御回路26に与えられ、メモリ制御回路26によってSDRAM28に書き込まれる。つまり、CCDイメージヤ12がプログレッシブスキャン方式を採用する一方、モニタ40はインタースキャン方式を採用するため、走査方式の変換のためにYUVデータが一時的にSDRAM28に格納される。格納されたYUVデータはその後、同じメモリ制御回路26によってインタースキャン方式で読み出され、バス24を介してビデオエンコーダ38に与えられる。ビデオエンコーダ38は、入力されたYUVデータからNTSCフォーマットに沿ったコンポジット映像信号を生成し、生成したコンポジット映像信号をモニタ40に入力する。この結果、被写体像の動画像が、リアルタイムでモニタ40に表示される。

【0015】オペレータがシャッターボタン48を操作すれば、被写体の動画像がメモリカード36に記録される。具体的には、オペレータがシャッターボタン48を1回押すと、CPU32がJPEGコーデック30に圧縮命令を与える。これに応じて、JPEGコーデック30は、メモリ制御回路26にYUVデータの読み出しリク

エストを出力するとともに、メモリ制御回路26によって読み出されたYUVデータにJPEG圧縮を施す。メモリ制御回路26は、読み出しリクエストに応じて1フレーム分のYUVデータつまり1画面分の静止画像データを読み出し、JPEGコーデック30に与える。このため、JPEGコーデック30では、1回の圧縮処理によって1画面分の圧縮画像データ（静止画像ファイル）が生成される。JPEGコーデック30は、このようにして生成した静止画像ファイルを、書き込みリクエストとともにメモリ制御回路26に与える。したがって、静止画像ファイルもまた、SDRAM28に一時的に格納される。

【0016】オペレータによってシャッタボタン48が再度押されない限り、CPU32はJPEGコーデック30に圧縮命令を繰り返し与える。これによって、複数の静止画像ファイルが生成され、SDRAM28に蓄積されていく。つまり、SDRAM28内に作成された1つの動画像ファイルに、静止画像ファイルが順次収納されていく。シャッタボタン48が押されると、CPU32は圧縮命令の出力を中止するとともに、メモリ制御回路26に動画像ファイルの読み出しリクエストを与える。この結果、静止画像ファイルの生成が終了されたとともに、SDRAM28から動画像ファイルが読み出される。読み出された動画像ファイルは、バス24およびI/F回路34を介して、メモリカード36に記録される。なお、複数画面分の静止画像データに順次JPEG圧縮を施して1つの動画像ファイルを作成する手法は、モーションJPEGと呼ばれる周知の手法である。

【0017】モード切換スイッチ50が“再生”側に切り換えられると、システムコントローラ42によって再生モードが設定される。すると、CPU32はメモリカード36に記録された動画像ファイルからヘッダを読み出し、メモリ制御回路26に書き込みリクエストを与える。ヘッダにはJPEG圧縮された先頭フレームのサムネイル画像データが含まれており、このような圧縮サムネイル画像データがメモリ制御回路26によってSDRAM28に書き込まれる。圧縮サムネイル画像データは、その後JPEGコーデック30によって伸長され、伸長されたサムネイル画像データもまた、SDRAM28に格納される。このような処理が9回繰り返され、9つの動画像ファイルに関連するサムネイル画像データがSDRAM28に確保される。一方、ビデオエンコーダ38は、メモリ制御回路26に対してサムネイル画像データの読み出しをリクエストする。メモリ制御回路26は、サムネイル画像データをインタースキャン方式でSDRAM28から読み出し、この結果、9つのサムネイル画像がモニタ40に表示される。つまり、各動画像の先頭フレーム画像が、サムネイル形式でモニタ40に表示される。

【0018】ここで、オペレータがカーソルキー44a

～44dおよびセットキー46を操作して、所望のサムネイル画像を選択すると、CPU32は、メモリカード36から対応する動画像ファイルを読み出すとともに、メモリ制御回路26に書き込みリクエストを与える。読み出された動画像ファイルは、SDRAM28に格納される。CPU32はまた、SDRAM28に格納された動画像ファイルをスキャンし、この動画像ファイルに収納された各静止画像ファイルの開始アドレスを検出する。そして、図2に示す動画像テーブル32aに、

“0”から始まるフレーム番号と検出したそれぞれの開始アドレスとを1対1対応で書き込んでいく。CPU32は、このような動画像テーブル32aを参照して、JPEGコーデック30に先頭の静止画像ファイルの伸長処理を命令する。JPEGコーデック30は、伸長命令が1回与えられるごとに所望の静止画像ファイルの読み出しリクエストをメモリ制御回路26に与え、読み出された静止画像ファイルにJPEG伸長を施す。

【0019】伸長が完了すると、JPEGコーデック30は、伸長された静止画像データを書き込みリクエストとともにメモリ制御回路26に与える。このため、静止画像データがSDRAM28に書き込まれる。ビデオエンコーダ38は、このような静止画像データの読み出しをメモリ制御回路26にリクエストし、この結果、対応する静止画像がモニタ40に表示される。つまり、所望のサムネイル画像が選択された直後は、サムネイル画像と同じ内容の静止画像がモニタ40の全面に表示される。

【0020】オペレータがセットキー46を押すと、CPU32は、SG16から出力される垂直同期信号に応答してJPEGコーデック30に次の静止画像ファイルの伸長処理を命令する。このときも、動画像テーブル32aを参照して、伸長する静止画像ファイルが特定される。JPEGコーデック30は、メモリ制御回路26に静止画像ファイルの読み出しをリクエストし、読み出された静止画像ファイルを伸長する。伸長処理によって得られた静止画像データもまた、一時的にSDRAM28に書き込まれ、その後ビデオエンコーダ38に入力される。このように、セットキー46が押されると、残りの静止画像ファイルが垂直同期信号に応答してSDRAM28から再生される。伸長された静止画像データは1フレーム期間ごとに生成され、モニタ40には、先頭の静止画像に続く動画像が表示される。

【0021】動画像が通常再生されている途中で左向きのカーソルキー44bが押されると、CPU32は、伸長処理が1回終了する毎に次の静止画像ファイルの伸長命令を出力する。つまり、通常再生の指示が与えられたとき、JPEGコーデック30は垂直同期信号に応答して伸長処理を行うが、カーソルキー44bが操作されると、JPEGコーデック30は伸長処理を継続して実行する。この結果、各静止画像ファイルは通常再生時より

も速いタイミングで伸長され、モニタ40に表示される静止画像は、通常再生時よりも速いタイミングで更新される。つまり、動画像が高速で再生される。カーソルキー44bが再度押されると、CPU32は垂直同期信号に応答して伸長命令を出力する。動画像の再生速度は、通常速度に戻される。このように、動画像の再生速度は、カーソルキー44bの操作に応答して切り替えられる。

【0022】動画像が順方向に再生されている途中で右向きのカーソルキー44aが押されると、動画像が逆方向に再生される。順方向の再生速度が通常速度および高速のいずれであっても、カーソルキー44aが押されると、動画像は逆方向に通常速度で再生される。逆方向への再生が開始された後は、カーソルキー44aが押されるごとに、再生速度が通常速度と高速とで切り換わる。なお、動画像が逆方向に再生されている途中で左向きのカーソルキー44bが押されたときも、上述と同様に再生方向が反転され、通常速度で再生が行われる。

【0023】動画像の再生中にインデックス生成ボタン52が操作されると、CPU32はインデックスファイルを生成する。CPU32はまた、現時点でモニタ40に出力されている静止画像データからサムネイル画像データを作成し、作成したサムネイル画像データを圧縮命令とともにJPEGコーデック30を与える。この結果、サムネイル画像データがJPEG圧縮される。CPU32は、図2に示す動画像テーブル32aから対応するフレーム番号を検出し、検出したフレーム番号とJPEG圧縮によって生成された圧縮サムネイル画像データとをインデックスファイルに収納する。サムネイル画像が収納されると、CPU32はインデックスファイルをメモリカード36に記録する。一方、モニタ40には、このサムネイル画像に対応する静止画像、つまりインデックス生成ボタン52の操作時点における静止画像が、継続して表示される。

【0024】なお、インデックスファイルは、図4に示すフォーマットで形成される。各サムネイル画像は、フレーム番号、データ長および圧縮サムネイル画像データを1組として、作成順に連なる。動画像の再生中にインデックス表示ボタン54が操作されると、CPU32は、メモリカード36からインデックスファイルを再生し、図3に示すようなサムネイル番号とフレーム番号とが対応付けられたインデックステーブル32bを作成する。サムネイル番号は、対応するサムネイル画像がインデックスファイルの先頭から何番目に収納されているかを示す番号である。CPU32はまた、サムネイル番号が“0”から“8”的圧縮サムネイル画像データをインデックスファイルから読み出すとともに、メモリ制御回路26に書き込みリクエストを与える。読み出された圧縮サムネイル画像データは、メモリ制御回路26によってSDRAM28に書き込まれる。

【0025】圧縮サムネイル画像データは、その後JPEGコーデック30によって伸長され、伸長されたサムネイル画像データもまた、SDRAM28に書き込まれる。伸長サムネイル画像データはビデオエンコーダ38に出力され、この結果、9つのサムネイル画像が図5に示す要領でモニタ40に再生される。なお、インデックスファイルに収納されたサムネイル画像が10以上あるときは、スクロール指示に応じて他のサムネイル画像がモニタ40に再生される。

【0026】オペレータがモニタ表示の中から所望のサムネイル画像を選択すると、CPU32は、対応するフレーム番号をインデックステーブル32bから検出す。検出したフレーム番号を持つ所望の静止画像ファイルは、SDRAM28に保持された動画像ファイルに収納されている。CPU32は、所望の静止画像ファイルの伸長をJPEGコーデック30に命令し、所望の静止画像ファイルが伸長される。これによって得られた静止画像データは、SDRAM28を介してビデオエンコーダ38に出力される。この結果、選択されたサムネイル画像と同じ内容の静止画像が、モニタ40の全面に表示される。なお、この後にオペレータがセットキー46を押せば、表示された静止画像に続く動画像が再生される。

【0027】再生モードが設定されたときのCPU32の処理動作を、図8～図19に示すフロー図を用いて説明する。CPU32はまずステップS1で、メモリカード36に記録されている9つの動画像ファイルからヘッダを検出し、検出されたヘッダに含まれるサムネイル画像データを再生する。ヘッダに含まれるサムネイル画像データは動画像ファイルの最初の静止画像ファイルに対応し、モニタ40には各動画像の先頭フレームのサムネイル画像が表示される。ステップS3で所望のサムネイル画像が選択されると、CPU32はステップS5で、メモリカード36から対応する動画像ファイルを読み出し、図2に示す動画像テーブル32aを作成し、さらにフレーム番号frmをカウントするカウンタ32cをリセットする。CPU32はまた、ステップS9でメモリ制御回路26に書き込みリクエストを出力する。この結果、読み出された動画像ファイルがSDRAM28に書き込まれる。

【0028】ステップS9では、動画像ファイルの所定フレーム目に収納された静止画像ファイルの伸長をJPEGコーデック30に命令する。JPEGコーデック30は、この伸長命令に応答してメモリ制御回路26に所定の静止画像ファイルの読み出しをリクエストし、読み出された静止画像ファイルを伸長する。JPEGコーデック30はまた、伸長された静止画像データの書き込みをメモリ制御回路26にリクエストし、静止画像データは再度SDRAM28に格納される。この静止画像データはビデオエンコーダ38からの読み出しリクエストに

応じて読み出され、この結果、モニタ40の全面に所定の静止画像が表示される。なお、ステップS5でカウント値frmがリセットされるため、ステップS5に続くステップS9の処理では、動画像ファイルの先頭に収納された静止画像ファイルの伸長が命令される。したがって、モニタ40には先頭の静止画像が表示される。

【0029】CPU32は続いて、ステップS11で所定の静止画像ファイルの伸長処理が終了したかどうか判断し、“YES”であれば、ステップS13でセットキー46が押されたかどうか判断する。オペレータがセットキー46を操作すれば、CPU32はステップS13で“YES”と判断し、ステップS17で再生速度を示すWフラグ32aおよび再生方向を示すDフラグ32bをリセットする。Wフラグ32aのリセット状態およびセット状態は、それぞれ通常速度および高速を意味し、Dフラグ32bのリセット状態およびセット状態は、それぞれ順方向および逆方向を意味する。ステップS19では、Wフラグ32aの状態を判別する。Wフラグ32aがセット状態であればそのままステップS23に進むが、リセット状態であれば、ステップS21を介してステップS23に進む。ステップS21では垂直同期信号の入力の有無を判断し、入力有りと判断されたときにステップS23に進む。

【0030】セットキー46の操作に応答してWフラグ32aがリセットされるため、セットキー46が押された直後は垂直同期信号の入力を待ってステップS23に進む。ステップS23では、Dフラグ32bの状態を判別する。そして、リセット状態であれば図9のステップS25に進むが、セット状態であれば図10のステップS47に進む。

【0031】Dフラグ32bがリセット状態のとき、CPU32は、ステップS25で次の静止画像ファイルの伸長をJPEGコーデック30に命令するとともに、ステップS27でカウンタ32cをインクリメントする。JPEGコーデック30は、CPU32からの伸長命令に応答して、上述と同じ要領で次の静止画像ファイルを読み出し、伸長処理を施す。この結果、モニタ40に次の静止画像が表示される。

【0032】CPU32は、JPEGコーデック30の伸長処理が1回終了すると、ステップS29で“YES”と判断する。続くステップS31では、伸長された静止画像ファイルが動画像ファイルの末尾に収納されたファイルであるかどうか判断する。末尾ファイルであれば、ステップS33でカウンタ32cをリセットしてからステップS9に戻る。この結果、先頭の静止画像ファイルが再度再生される。つまり、動画像が最後まで再生されると、モニタ40の表示は先頭の静止画像に戻る。一方、末尾ファイルでなければ、CPU32はステップS35およびステップS43のそれぞれでカーソルキー44bおよび44aの操作の有無を判別する。

【0033】カーソルキー44bが操作されれば、CPU32はステップS37で“YES”と判断し、ステップS37～S41でWフラグ32aの状態を反転させる。つまり、ステップS37でWフラグ32aの状態を判別し、リセット状態であればステップS39でWフラグ32aをセット状態に切り換えるが、セット状態であればステップS41でWフラグ32aをリセット状態に切り換える。そして、ステップS19に戻る。カーソルキー44aが操作されたときは、CPU32はステップS43で“YES”と判断し、ステップS45でWフラグ32aおよびDフラグ32bの両方をリセットする。そして、ステップS19に戻る。

【0034】ステップS43で“NO”であれば、CPU32は、ステップS69およびステップS73のそれぞれでインデックス生成ボタン52およびインデックス表示ボタン73が押されたかどうか判断する。インデックス生成ボタン52が押されれば、ステップS71でインデックス生成処理を実行してからステップS9に戻るが、インデックス表示ボタン54が押されれば、ステップS75でインデックス表示処理を実行してからステップS9に戻る。なお、カーソルキー44aおよび44b、インデックス生成ボタン52ならびにインデックス表示ボタン54のいずれも操作されなければ、CPU32はそのままステップS19に戻る。

【0035】図8のステップS23で“YES”と判断されると、CPU32は図10のステップS47以降の処理を実行するが、ステップS47～S67の処理は、ステップS47で先行する静止画像ファイルの伸長を命令する点、ステップS49でカウンタ32cをディクリメントする点、ステップS53で伸長された静止画像ファイルが先頭ファイルであるかどうかを判別する点、ならびにステップS57およびS65でカーソルキー44aおよび44bの操作の有無をそれぞれ判別する点を除き、図9に示すステップS25～S45の処理と同様である。このため、重複する部分についての説明を省略する。

【0036】CPU32は、ステップS5において図12に示すサブルーチンを処理する。まずステップS101で所望の動画像ファイルをメモリカード36から読み出し、次にステップS103でメモリ制御回路26に書き込みリクエストを与える。この結果、読み出された動画像ファイルはSDRAM28に格納される。ステップS105では、SDRAM28に格納された動画像ファイルをスキャンして、図5に示す動画像テーブル32aを作成する。動画像テーブル32aには、先頭から順に割り当てられたフレーム番号と各静止画像ファイルの開始アドレスとが書き込まれる。その後、ステップS107でカウンタ32cをリセットし、図8のメインルーチンに復帰する。

【0037】図11のステップS71では、図13に示

すサブルーチンが処理される。CPU32はまずステップS201で、現動画像ファイルに対応するインデックスファイルがメモリカード36に存在するかどうか判断する。つまり、インデックスファイルは、オペレータの指示に応じて各画像ファイルに1つずつ作成される。このため、ステップS201で現動画像ファイルに対応するインデックスファイルの有無が判別される。インデックスファイルが存在しなければ、CPU32はステップS207でインデックスファイルを新規に作成し、ステップS209に進む。一方、インデックスファイルが存在するのであれば、ステップS202でインデックスファイルをメモリカード36から読み出し、ステップS203でインデックスファイルに書き込まれたフレーム番号をスキャンする。ステップS205では、現フレーム番号と同じ番号がインデックスファイルに存在するかどうか判断する。そして、“YES”であればそのままメインルーチンに復帰するが、“NO”であればステップS209に進む。

【0038】ステップS209では、現フレームの静止画像データつまり現時点モニタ40に出力されている静止画像データに間引き処理を施して、サムネイル画像データを生成する。その後、ステップS211でJPEGコーデック30にサムネイル画像データの圧縮命令を与える。JPEGコーデック30によって圧縮されたサムネイル画像データは、メモリ制御回路26によってSDRAM28に格納される。圧縮処理が完了すると、CPU32はステップS213で“YES”と判断し、ステップS215でメモリ制御回路26に読み出しリクエストを与える。これに応じて、SDRAM28から圧縮サムネイル画像データが読み出される。ステップS216では動画像テーブル32aを参照して現フレーム番号を検出し、検出した現フレーム番号をカウンタ32cにセットする。また、ステップS217では、読み出された圧縮サムネイル画像データを現フレーム番号とともにインデックスファイルに収納する。そして、ステップS219でインデックスファイルをメモリカード36に記録し、メインルーチンに復帰する。

【0039】現フレーム番号がカウンタ32cにセットされることで、図8のステップS9では、記録されたサムネイル画像に対応する静止画像ファイルが再生される。そして、セットキー46が押されない限り、同じ静止画像が継続してモニタ40に表示される。セットキー46が押されると、この静止画像に続く動画像が再生される。

【0040】ステップS75では、図14に示すサブルーチンが処理される。CPU32は、ステップS301で現動画像ファイルに対応するインデックスファイルがメモリカード36に存在するかどうか判断する。ここで“NO”であればそのままメインルーチンに復帰するが、“YES”であればステップS303でインデック

スファイルをメモリカード36から読み出す。続いて、ステップS305でインデックスファイルをスキャンし、図6に示すインデックステーブル32bを作成する。ステップS307では、図15に示すサブルーチンに従って複数のサムネイル画像をモニタ40に表示し、ステップS309では図16～図19に示すサブルーチンに従ってサムネイル画像を選択する。

【0041】その後ステップS311で、選択されたサムネイル画像のフレーム番号をインデックステーブル32bから検出し、検出したサムネイル番号をカウンタ32cにセットする。そして、メインルーチンに復帰する。この結果、図8のステップS9では、選択されたサムネイル画像に対応する静止画像ファイルの伸長命令がJPEGコーデック30に与えられ、選択されたサムネイル画像と同じ静止画像がモニタ40の全面に表示される。この静止画像の表示は、セットキー46が押されない限り継続される。そして、セットキー46が押されたと、この静止画像に続く動画像が再生される。

【0042】図15を参照して、CPU32は、ステップS401でカウンタ32d～32fをリセットする。カウンタ32dはモニタ40に表示するサムネイル画像の番号（サムネイル番号）nをカウントするカウンタであり、カウンタ32dはモニタ40におけるサムネイル画像の表示位置番号ndをカウントするカウンタであり、そしてカウンタ32fはモニタ40におけるカーソルの表示位置番号nsをカウントするカウンタである。また、モニタ40には、図5に示すように9つのサムネイル画像が表示される。このため、サムネイル画像およびカーソルの表示位置には、図6および図7に示すように位置番号“0”～“8”が割り当てられる。なお、ステップS403では、インデックスファイルからn番目の圧縮サムネイル画像データを読み出し、同時にJPEGコーデック30に伸長命令を与える。この結果、n番目の圧縮サムネイル画像データが伸長され、伸長された画像データがSDRAM28に書き込まれる。モニタ40には、対応するサムネイル画像が表示される。JPEGコーデック30の伸長処理が完了すると、CPU32はステップS407で“YES”と判断し、ステップS409でカウント値nがインデックスファイルに格納されているサムネイル画像の総数Nに達したかあるいは伸長処理が9回行われたかを判断する。サムネイル画像の総数Nは、インデックステーブル32bを参照して算出される。

【0043】ステップS409で“NO”であればステップS403に戻り、上述の処理を繰り返す。ステップS403～S409の処理が繰り返されることで、複数のサムネイル画像がモニタ40に表示される。一方、ステップS409で“YES”であれば、ステップS411でカウント値nsをリセットとともにカウント値nから“9”を減算する。これによって、最初に表示さ

れたサムネイル画像が特定される。その後、ステップS 4 1 3でカウント値n sが示す位置にカーソルを表示し、図14のサブルーチンに復帰する。

【0044】この結果、インデックス表示ボタン5 4が押された直後は、図6に示す要領でサムネイル画像およびカーソルが表示される。つまり、インデックスファイルの先頭から9つのサムネイル画像が表示され、n s = 0に対応する位置にカーソルが表示される。なお、インデックスファイルに格納されているサムネイル画像が9つに満たなければ、全てのサムネイル画像が表示され
る。

【0045】図14に示すステップS 3 0 9では、図1 6～図1 9に示すサブルーチンが処理される。CPU 3 2は、ステップS 5 0 1, S 5 1 9, S 5 3 7およびS 5 5 5のそれぞれでカーソルキー4 4 b, 4 4 a, 4 4 dおよび4 4 cが押されたがどうかを判断し、ステップS 5 6 9でセットキー4 6が押されたかどうかを判断する。カーソルキー4 4 bが押されればステップS 5 0 3～S 5 1 7を処理し、カーソルキー4 4 aが押されればステップS 5 2 1～S 5 3 5を処理する。また、カーソルキー4 4 dが押されればステップS 5 3 9～S 5 5 3を処理し、カーソルキー4 4 cが押されればステップS 5 5 5～S 5 6 7を処理する。セットキー4 6が押されれば、ステップS 5 7 1を処理する。

【0046】まず図16を参照して、CPU 3 2は、ステップS 5 0 3でカウント値n sをインクリメントし、ステップS 5 0 5でカウント値n sを“9”と比較する。n sが“9”未満であれば、ステップS 5 0 7でサムネイル番号nおよびカーソル位置番号n sの和をサムネイル画像の総数Nと比較する。現時点のサムネイル番号nは図3の位置“0”に表示されているサムネイル画像の番号であり、n + n sはカーソル移動先のサムネイル番号である。n + n sがN未満であれば、カーソルを移動する余地が残っているため、現カウント値n sに対応する位置にカーソルを移動させる。たとえばn s = 1であれば、図3の位置“1”を指すようにカーソルが表示される。そして、ステップS 5 0 1に戻る。一方、n s + n sがN以上であればカーソルが移動する余地はないため、CPU 3 2は、ステップS 5 0 9でカウント値n sを元に戻してからステップS 5 0 1に戻る。

【0047】ステップS 5 0 5で“YES”的ときも、CPU 3 2はステップS 5 1 3でn + n sとNとを比較する。そして、“YES”であれば、ステップS 5 1 5でカウント値n sを元に戻してからステップS 5 0 1に戻る。一方、ステップS 5 1 3で“NO”であれば、ステップS 5 1 7で、現サムネイル番号に“3”を加算するとともに、現カーソル位置番号n sを“6”に更新する。そして、図15のステップS 4 0 3に移行する。n s ≥ 9でn + n s < Nの場合、カーソルが指そうとしているサムネイル画像は、インデックスファイルにはある

もののモニタ4 0には表示されていない。このため、モニタ4 0の表示をスクロールすべく、ステップS 5 1 7および図15のステップS 4 0 3以降が処理される。したがって、カーソルが図6に示すサムネイル画像8を指している状態でカーソルキー4 4 bが押されると、モニタ4 0の表示は図7に示すように更新され、カーソルはサムネイル画像9を指す。

【0048】図17を参照して、カーソルキー4 4 aが押された場合、CPU 3 2は、ステップS 5 2 1でカウント値n sつまりカーソル位置番号をディクリメントし、ステップS 5 2 3でカウント値n sが“0”に満たないかどうか判断する。n s ≥ 0であれば、CPU 3 2はステップS 5 3 5でカウント値n sが示す位置にカーソルを移動させ、ステップS 5 0 1に戻る。一方、n s < 0であれば、ステップS 5 2 5で現サムネイル番号nから“3”を引き算し、ステップS 5 2 7で引き算後のサムネイル番号nを“0”と比較する。n < 0であれば、カーソルに移動の余地はない。このため、CPU 3 2はステップS 5 3 1およびS 5 3 3でサムネイル番号nおよびカーソル位置番号n sを元に戻し、ステップS 5 0 1に戻る。ステップS 5 2 7でn ≥ 0であれば、ステップS 5 2 9でカーソル位置番号n sを“2”にセットし、ステップS 4 0 3に移行する。したがって、たとえばカーソルが図7に示すサムネイル画像3を指しているときにカーソルキー4 4 aが押されると、モニタ4 0の表示は図6に示すように更新され、かつカーソルはサムネイル画像2を指す。

【0049】図18を参照して、カーソルキー4 4 dが押されると、CPU 3 2は、ステップS 5 3 9で現カーソル位置番号n sから“3”を引き算し、ステップS 5 2 3で引き算後のカーソル位置番号n sが“0”に満たないかどうか判断する。n s ≥ 0であれば、CPU 3 2はステップS 5 5 3でカウント値n sが示す位置にカーソルを移動させ、ステップS 5 0 1に戻る。しかし、n s < 0であれば、ステップS 5 2 5で現サムネイル番号nから“3”を引き算し、ステップS 5 2 7で引き算後のサムネイル番号nを“0”と比較する。n < 0であれば、カーソルに移動の余地はないため、CPU 3 2はステップS 5 4 9およびS 5 5 1でサムネイル番号nおよびカーソル位置番号n sを元に戻す。そして、ステップS 5 0 1に戻る。ステップS 5 2 7でn ≥ 0であれば、ステップS 5 2 9でカーソル位置番号n sだけを元に戻し、ステップS 4 0 3に移行する。したがって、たとえばカーソルが図7に示すサムネイル画像4を指しているときにカーソルキー4 4 dが押されると、モニタ4 0の表示は図6に切り換わり、カーソルはサムネイル画像1を指す。

【0050】図19を参照して、カーソルキー4 4 cが押されると、CPU 3 2は、ステップS 5 5 7でカウント値n sに“3”を加算し、ステップS 5 5 9で加算後

のカウント値 n_s を “9” と比較する。ここで $n_s \geq 9$ であれば、CPU32は、ステップS567でカウント値 n に “3” を加算とともにカウント値 n_s を元に戻し、ステップS403に移行する。一方、 $n_s < 9$ であれば、CPU32はステップS561で $n + n_s$ を N と比較する。 $n + n_s < N$ であればカーソルに移動の余地が残っているため、CPU32は、ステップS565でカウント値 n_s が示す位置にカーソルを移動させ、ステップS501に戻る。これに対して $n + n_s \geq N$ であれば、カーソルはこれ以上移動できない。このため、CPU32は、ステップS563でカウント値 n_s を元に戻し、その後ステップS501に戻る。したがって、たとえばカーソルが図6に示すサムネイル画像7を指しているときにカーソルキー44cが押されると、モニタ40の表示は図7に更新され、カーソルはサムネイル画像4を指す。

【0051】セットキー46が押された場合、CPU32は、ステップS571でフレーム番号 f_{rm} を $n + n_s$ と決定し、図14に示すサブルーチンに復帰する。このため、ステップS311では $n + n_s$ がカウンタ32cにセットされ、図8に示すステップS9ではカーソルが指しているサムネイル画像に対応する静止画像ファイルに伸長処理が施される。この結果、サムネイル画像と同じ静止画像がモニタ40の全面に表示される。なお、ステップS569で “NO” であれば、ステップS501に戻る。

【0052】以上のように、SDRAM28に保持された動画像ファイルの再生中にインデックス表示ボタン54が押されると、メモリカード36からインデックスファイルが読み出され、これに収納されている複数のサムネイル画像がモニタ40に表示される。ここで、表示されているサムネイル画像の中から所望のサムネイル画像が選択されると、選択されたサムネイル画像に対応する静止画像ファイルが動画像ファイルから読み出され、選択されたサムネイル画像と同じ静止画像がモニタ40の全面に表示される。さらに、セットキー46が押されると、この静止画像に続く動画像が再生される。したがって、所望の記録部分に短時間でアクセスできる。

【0053】なお、この実施例では、モーションJPEGを利用して動画像を圧縮する場合について説明したが、動画像はMPEGフォーマットに従って圧縮するようにしてもよい。但し、MPEGの場合、データはストリーム構造となっており、基準となるフレーム画像は所定フレーム数おきに含まれる。このため、インデックスファイルに収納するサムネイル画像は、このような基準フレーム画像から作成する必要がある。

【0054】また、この実施例ではオペレータの指示に応じてサムネイル画像をインデックスファイルに収納するようにしたが、サムネイル画像はオペレータの指示に関係なく所定フレームおきに作成するようにしてもよ

い。また、この実施例ではデジタルカメラを用いて説明したが、この発明は、動画像ファイルを再生するあらゆる画像再生装置に適用できることは言うまでもない。

【図面の簡単な説明】

【図1】この発明の1実施例を示すブロック図である。

【図2】動画像テーブルの一例を示す図解図である。

【図3】インデックステーブルの一例を示す図解図である。

【図4】インデックスファイルのフォーマットの一例を示す図解図である。

【図5】モニタに表示されたサムネイル画像の例を示す図解図である。

【図6】モニタに表示されたサムネイル画像の他の例を示す図解図である。

【図7】モニタに表示されたサムネイル画像のその他の例を示す図解図である。

【図8】図1実施例の動作の一部を示すフロー図である。

【図9】図1実施例の動作の他の一部を示すフロー図である。

【図10】図1実施例の動作のその他の一部を示すフロー図である。

【図11】図1実施例の動作のさらにその他の一部を示すフロー図である。

【図12】図1実施例の動作の他の一部を示すフロー図である。

【図13】図1実施例の動作のその他の一部を示すフロー図である。

【図14】図1実施例の動作のさらにその他の一部を示すフロー図である。

【図15】図1実施例の動作の他の一部を示すフロー図である。

【図16】図1実施例の動作のその他の一部を示すフロー図である。

【図17】図1実施例の動作のさらにその他の一部を示すフロー図である。

【図18】図1実施例の動作の他の一部を示すフロー図である。

【図19】図1実施例の動作のその他の一部を示すフロー図である。

【符号の説明】

10 …デジタルカメラ

24 …バス

26 …メモリ制御回路

28 …SDRAM

30 …JPEGコーデック

32 …CPU

36 …メモリカード

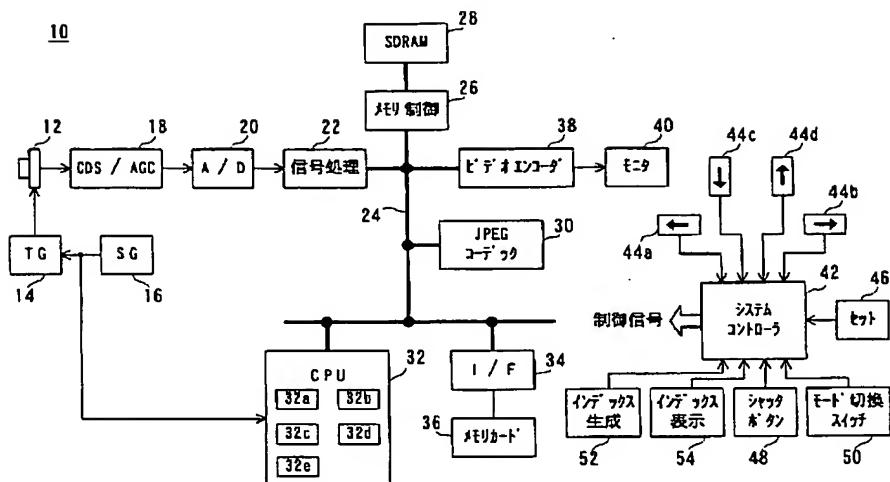
38 …ビデオエンコーダ

40 …モニタ

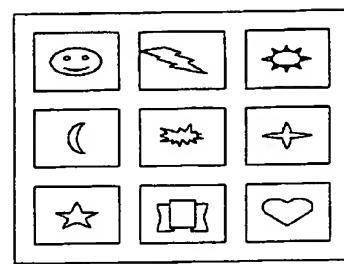
5.2 …インデックス生成ボタン

5.4 …インデックス表示ボタン

【図1】



【図5】



【図2】

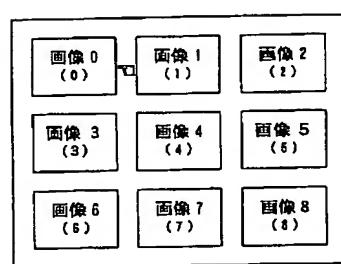
フレーム番号	静止画アドレス
0	-----
1	-----
2	-----
⋮	⋮
⋮	⋮
⋮	⋮
⋮	⋮

32a

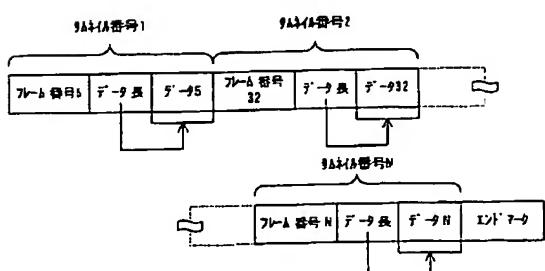
サムネイル番号	フレーム番号
1	5
2	32
3	64
4	87
5	250
⋮	⋮
⋮	⋮
⋮	⋮

32b

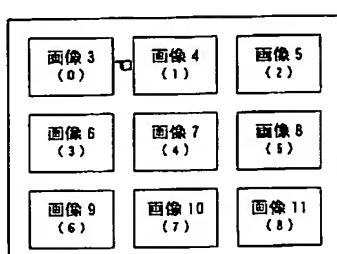
【図6】



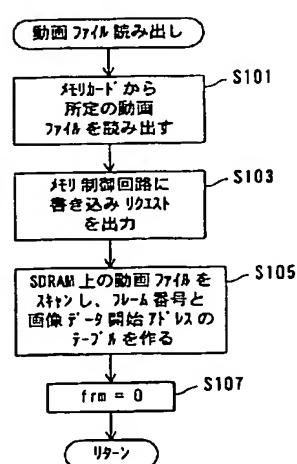
【図4】



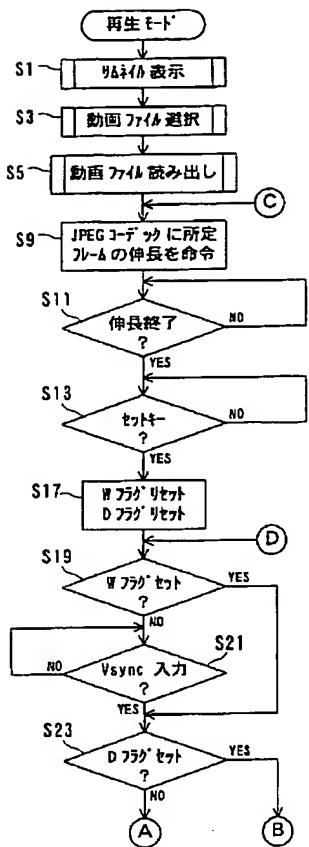
【図7】



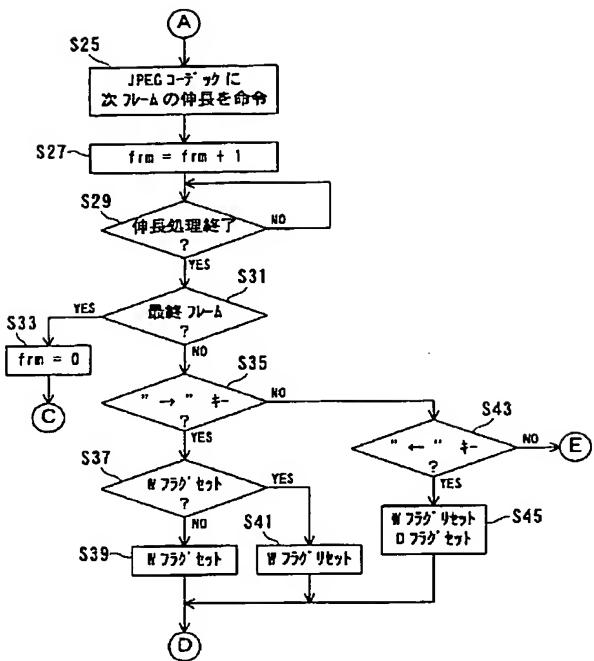
【図12】



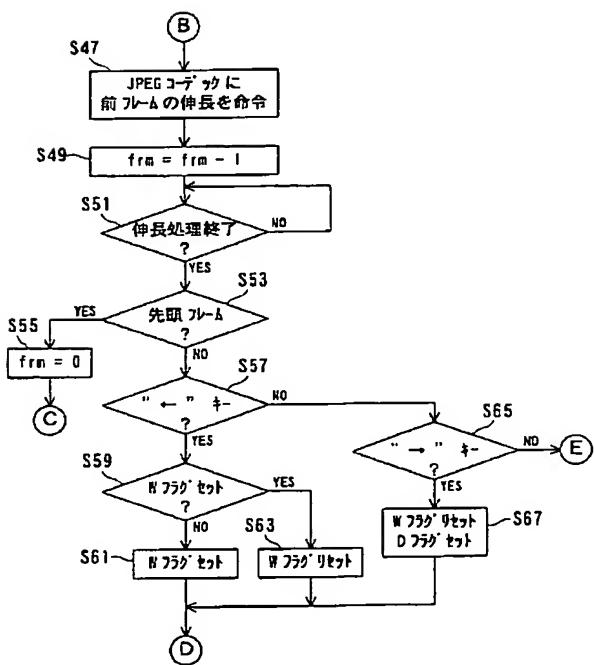
【図8】



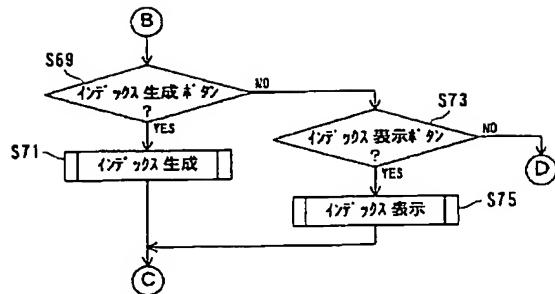
【図9】



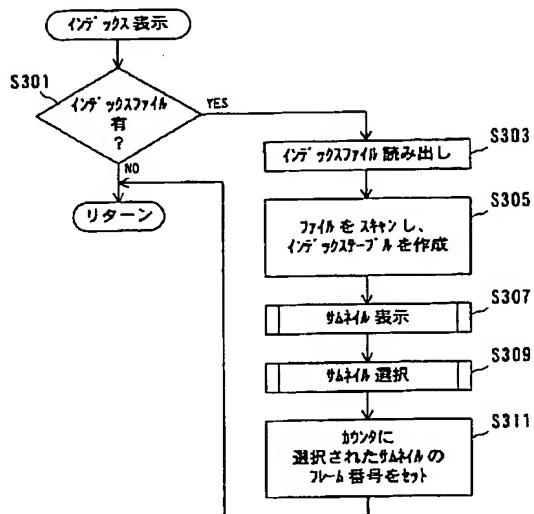
【図10】



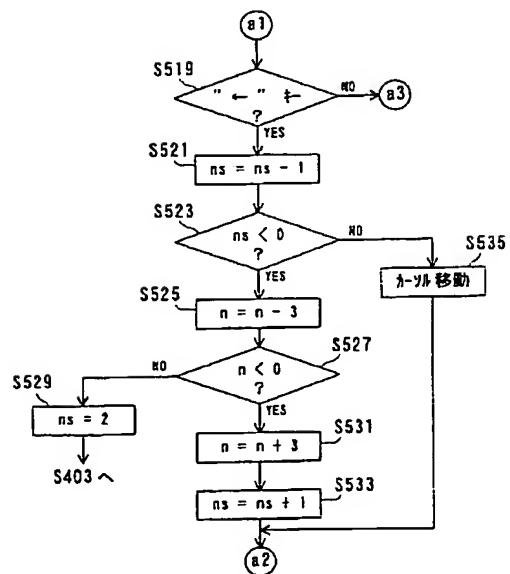
【図11】



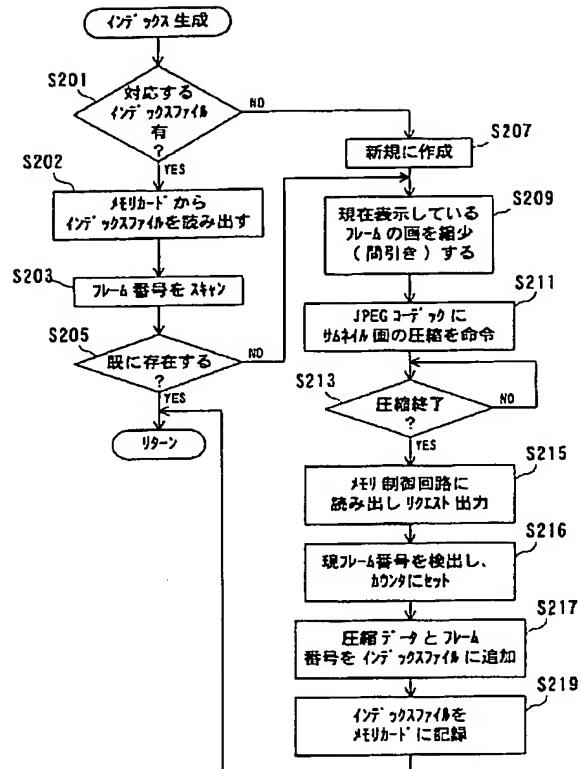
【図14】



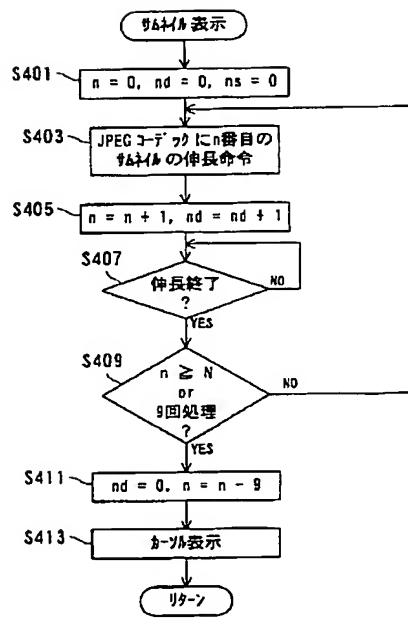
【図17】



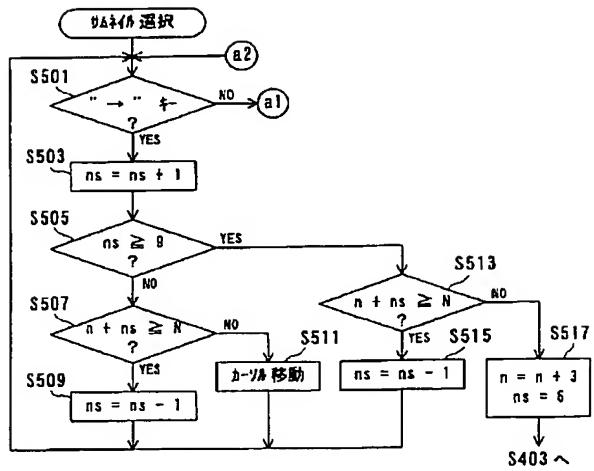
【図13】



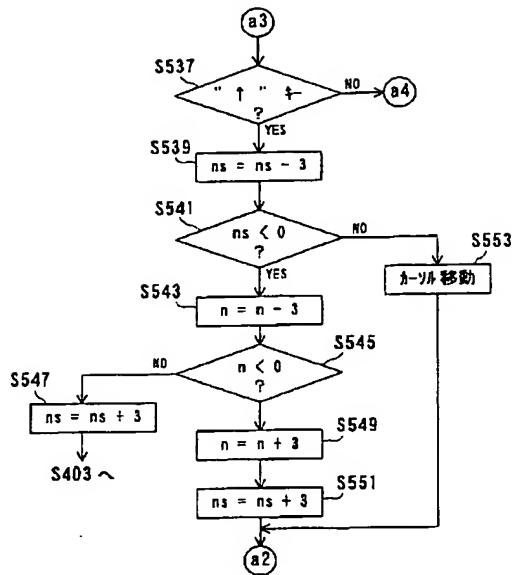
【図15】



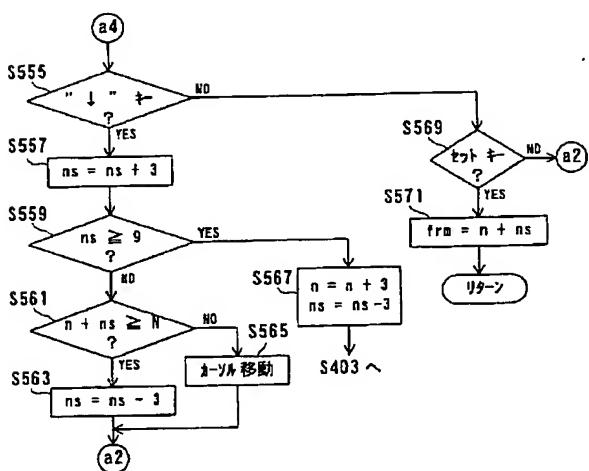
【図16】



【図18】



【図19】



**This Page is Inserted by IFW Indexing and Scanning
Operations and is not part of the Official Record**

BEST AVAILABLE IMAGES

Defective images within this document are accurate representations of the original documents submitted by the applicant.

Defects in the images include but are not limited to the items checked:

- BLACK BORDERS**
- IMAGE CUT OFF AT TOP, BOTTOM OR SIDES**
- FADED TEXT OR DRAWING**
- BLURRED OR ILLEGIBLE TEXT OR DRAWING**
- SKEWED/SLANTED IMAGES**
- COLOR OR BLACK AND WHITE PHOTOGRAPHS**
- GRAY SCALE DOCUMENTS**
- LINES OR MARKS ON ORIGINAL DOCUMENT**
- REFERENCE(S) OR EXHIBIT(S) SUBMITTED ARE POOR QUALITY**
- OTHER:** _____

IMAGES ARE BEST AVAILABLE COPY.

As rescanning these documents will not correct the image problems checked, please do not report these problems to the IFW Image Problem Mailbox.